Queensland University of Technology
Brisbane Australia

This is the author's version of a work that was submitted/accepted for publication in the following source:

Yulizar, Y., Yeung, K.H., Mattabadal, V., & Chandran, V. (1995) Implementation of a survey form marking system using image processing techniques on an Imputer. In *3rd Conference on Digital Imaging Computing Techniques and Applications (DICTA 95)*, 6-8 December 1995, University of Queensland, Brisbane, QLD.

This file was downloaded from: http://eprints.qut.edu.au/46491/

© Copyright 1995 [Please consult authors]

Notice: *Changes introduced as a result of publishing processes such as copy-editing and formatting may not be reflected in this document. For a definitive version of this work, please refer to the published source:*

Implementation of a Survey Form Marking System using Image Processing Techniques on an Imputer

Y. Yulizar, K. H. Yeung, V. Mattabadal and V. Chandran

Signal Processing Research Centre,
Queensland University of Technology,
2 George Street, Brisbane, 4001, QLD.

ABSTRACT

This paper describes the feasibility of the application of an Imputer[1] in a multiple choice answer sheet marking system based on image processing techniques. Unlike the marking machine currently used in the Queensland University of Technology (QUT) which only accepts the forms supplied by its manufacturer, the Imputer-based marking system can accept answer sheets of user designed formats. The computation time is proportional to the number of questions to be marked, ranging from 4 seconds for 20 questions to 10 seconds for 100 questions, and can be improved if the system is fitted with an automatic sheet feeder. The interactive window-based program accompanying the system outputs the answers of the sheet being marked as well as the details of the student. The parameters of the program can be easily changed to take into account the differences between different types of answer sheets. The program uses adaptive thresholding to adjust to different lighting conditions and different strengths of pencil marks. The system, comprising a personal computer, an Imputer and a light source, can be easily assembled and disassembled so that the hardware can be used for other applications when not performing any marking job.

1. INTRODUCTION

A marking machine can be treated as a data-input device. The input data is in the form of pencil marks on formatted sheets of paper, representing answers to multiple choice questions as well as names and identification numbers. It needs to be converted into appropriate alphanumerals for storage and further processing by a digital computer. A marking machine performs this task with little or no operator intervention for data entry. The criteria which the design of a marking system has to be met are accuracy, efficiency, and flexibility.

The marking machine used in QUT is an OpScan[2] 7 scanner. It is a desktop optical mark reading (OMR) scanner designed for a wide variety of applications in the education environment such as grade reporting, test scoring and survey processing. It is fast, requires minimum operator intervention and can read two-sided documents. However, the apparent limitations of its application are: (1) it can only scan the standard sheets provided by its manufacturer; (2) the computation time for each answer sheet is fixed regardless of the number of questions to be marked; (3) the machine will become idle whenever it is not performing any marking job; (4) it is not cost effective for small-scale marking jobs.

[1] Imputer is trademark of VLSI Vision Ltd.
[2] Opscan is trademark of National Computer Systems.

In this paper, we present the design of an Imputer-based marking system based on image processing techniques. The algorithm is based on an adaptive thresholding approach to accurately locate the marked circles on the answer sheet. The proposed system has been successfully implemented on the Imputer and it has the following desirable features: (1) it is accurate and reliable; (2) the computation time decreases if only fewer questions are to be marked; (3) it is flexible, meaning that different formats of answer sheet can be accepted provided that some parameters of the program are calibrated accordingly; (4) it is cost effective for small to medium scale marking jobs; (5) it can be easily assembled and disassembled and the Imputer can be programmed for other applications.

2. IMPUTER

An Imputer is a completely programmable camera based on a new approach of combining image sensing with control functions on a single CMOS chip [1]. Compared with the multi-chip CCDs vision systems, it is smaller, cheaper and consumes less power. It replaces a camera, frame grabber, processing board and PC with a single integrated architecture. The main components of an Imputer consist of a motherboard with a built-in 312x287 image sensor, an 8032 microcontroller, 128 Kbytes RAM, 128 Kbytes flash memory, an A/D converter, a field-programmable gate array (FPGA) and an interface card. In normal operation, the sensor performs auto-exposure and auto-gain control. The FPGA provides all the necessary bus interface logic and control logic for the frame grabber. The interface card serves as a platform for all the physical connectors as well as providing some buffer logic. The motherboard communicates with a PC via the interface card [9].

The Imputer is supplied with a window-based software package known as the Imputer Development System (IDS) which enables the user to analyse images, build and download applications, upload data from the Imputer, check the hardware and view the RS232 output. A full library of machine-vision functions is provided, including transforms, convolvers, correlators, morphological filters, frequency filtering, image segmentation and logical operators [10]. Once an image has been captured and analysed, the Imputer can interact with its environment using the RS232 interface and the eight binary I/Os. The computer is only needed during application development.

3. ALGORITHM

The algorithm first converts the 8-bit greyscale image of the answer sheet where picture elements (pixels) have intensities in the range 0-255 to a bilevel image in which marked pixels are black and the other pixels white. The main issues here are the thresholding and the decision making required to determine which pixels are marked black. The algorithm must then decide which circles are marked. A circle is treated as marked if the number of marked pixels enclosed within it exceeds a preset value. Finally, the algorithm must determine the locations of the marked circles because the answer sheet is formatted and different locations correspond to different alphabets or numerals. Thus, the most important parts of the algorithm deal with:

- how to set an adaptive threshold level for the image
- how to locate a marked circle in the threshold image

Thresholding is an effective pre-processing tool in this application which reduces the computation time and the complexity of the algorithm. There are two major groups of thresholding techniques, namely, the local and global methods [4]. The global techniques are preferable in this algorithm owing to their low computational complexity and consequently higher speed.

The simplest and most efficient technique, with regards to speed of computation and insensitivity to object form, is the grey-level histogram global thresholding [4]. However, it is found that the distribution of the intensity level of the image of a marked answer sheet has a finite spread as shown in Figure 1 and it varies slightly. This makes the use of histogram in determining a global threshold level inappropriate to the application.

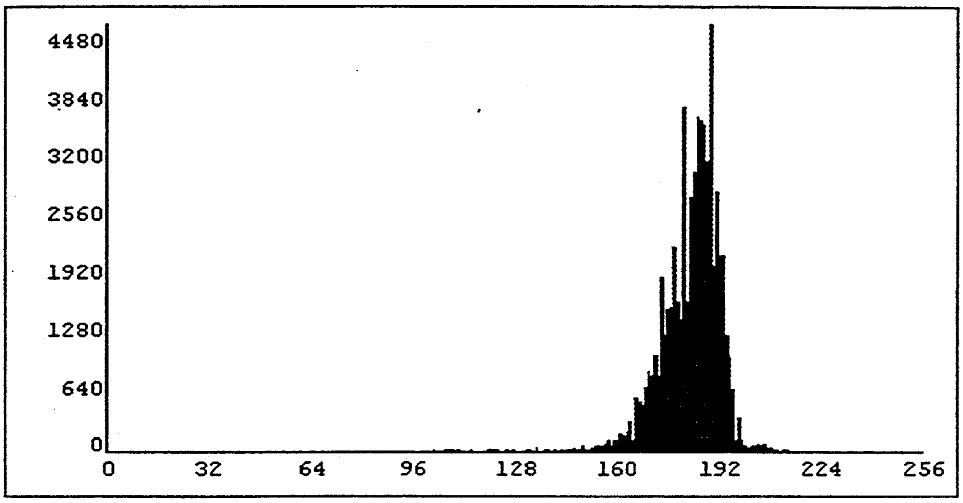
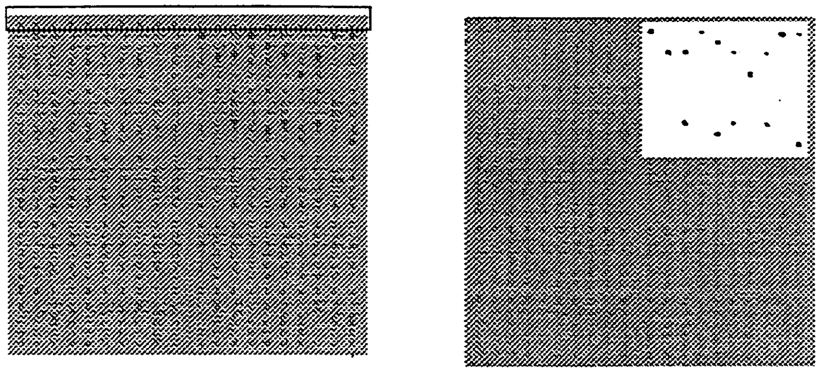


Figure 1. Histogram of a marked answer sheet.

Several other thresholding techniques were considered to tackle this problem. For instance, a method suggested in [8] is based on the assumption that the overall distribution function is a bimodal density. The threshold value is selected based on the means of the two brightness levels. This method was not effective for this application either since the desired separation could not be accomplished (see Figure 1). Other techniques as discussed in [2,3,5] were found to be computationally inefficient.

The difficulty arises from greylevel differences between different regions of the sheet owing to different illumination of the surface, and variations in the greylevels of the alphanumerals already printed on the sheet using a light colour (such as question numbers). An adaptive thresholding strategy has been designed to compensate for this. The first step is to select a sample area where the minimum level of the background (the whole sheet excluding the pencil marks) possibly lies. This area depends on the type of the form and is chosen at the time of calibration by processing a blank sheet (without any pencil marks). For the form chosen in this example the strip area which covers question numbers along the top most row (question number 1-10 and 51-100), as shown in Figure 2a, is found to be the darkest. In order to avoid the noise effect on the minimum value, a 3x3 averaging mask is run over this region to find the minimum average value. The threshold level is set as the minimum average value minus an offset. The offset is found to be 15 grey level experimentally. This method is found to be fast and robust while maintaining a high level of accuracy. Figure 2b shows the answer part of an answer sheet with the first 20 questions threshold. This thresholding is more effective than a single global threshold in taking into account greylevel variations from region to region. At the same time, it is less sensitive to noise and computationally more efficient than a local adaptive threshold computed from grey levels over a small window.



(a) (b)

Figure 2. (a) Original image of answer section, the area covered by the box is used to determine the threshold level ; (b) Thresholded image.

To detect an answer, the maximum number of marked pixels in one answer circle is obtained by running a 3×3 mask over the circle and the number of marked pixels in the mask is added in each run. If this number is equal to or greater than 4 in any run, the circle is accepted as being marked. The reason for choosing a level of 4 is that, after several trials it was found that any noise caused at most two pixels in a circle (3x3 neighbourhood) to cross the threshold. In other words, the signal-to-noise ratio of the binary image after thresholding is better than 2/9 or 22% in terms of the percentage of pixels affected by salt-and-pepper noise. The probability of four pixels appearing as marked because of noise is extremely low and could not be detected in the trials conducted. This guaranteed that the result has no false detection on any blank answer. In reality, the probability of false detection is extremely low. The acceptance criteria for partially filled answers can be adjusted by changing the number of marked pixels required to decide whether a circle is marked or not.

4. RESULTS

The system is set up in such a way that the Imputer views the answer sheets sitting in a fixed tray from above. The background lighting was provided by normal fluorescent light and the minimum of the brightest intensity measured from the sheets was found to be around grey level 198. The forms used were the standard ones currently used at QUT for multiple choice answer sheets, namely the SC-SP 1738 form supplied by ScanForm Corporation Pty Ltd . A 12 mm lens was used on the Imputer. The lens allows the answer section on the sheet to fit the size of the 256×256 pixels image. Therefore, each sheet was imaged twice starting from the student information section and then followed by the answers section.

Initially, a calibration stage was carried out to determine the parameters of the type of the form to be processed. The parameters consist of the reference point coordinate of the form and the distance between the centers of two adjacent answer circles both columnwise and rowwise. These parameters were obtained by capturing the image of the sample form and found their values manually. For the form used in the experiment the center of top-rightmost circle is chosen as the reference point.

The system was continuously tested on batches of 10 sheets. Batches of 10 at a time is recommended so that the images are always in focus. Different filled patterns (such as half-filled, quarterly-filled, cross, dot, lightly-marked) were tried to test the capability of the system. The questions were marked with single answer and multiple answers on the same sheet and 2B pencils were used throughout the tests. The results showed that 100% detection can be achieved provided: (1) There are at least 4 neighbouring marked pixels in the answer circle. (2) The grey level of the marked pixel is lower than the threshold value. In other words, a partly-filled circle will also be recognised. An example display showing the results is given in Figure 3.

The diameter of one answer circle is 3 mm (or 6 pixels) and the clearance between adjacent circles in the same question is 1mm (or 2 pixels) . Hence a maximum tolerance of 1 mm is allowed for setting up the tray to prevent false detection.
The average computation times for marking sheets of 20, 50 and 100 questions (excluding the time required to reposition the sheets, which is user dependent) were recorded to be 4, 6 and 10 seconds respectively.

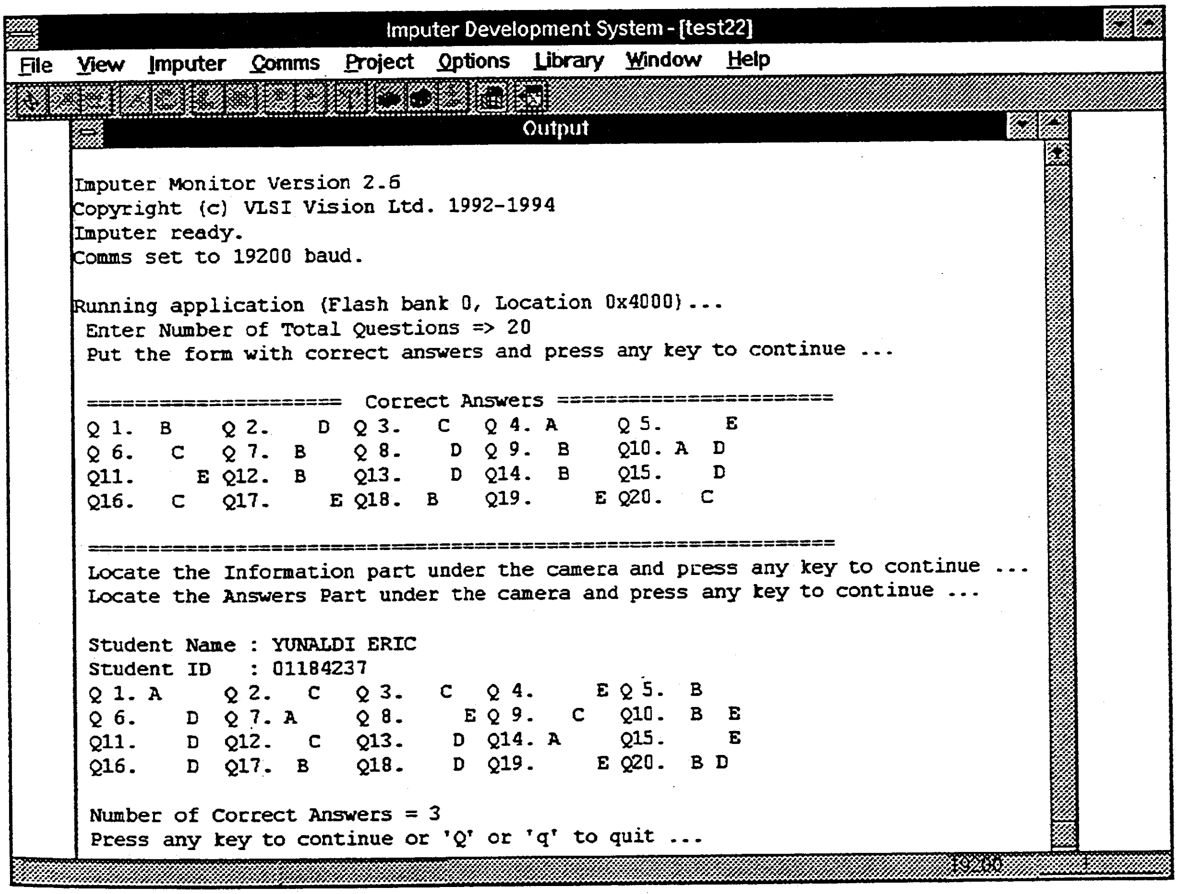


Figure 3. Display of results

5. CONCLUSIONS

The system was tested to have a 100% accuracy without regard to single or multiple answers provided that the answers were marked according to the format illustrated on the answer sheet. It also allows partly-filled answers. Although the testing model was built in such a way that it has to be operated manually, it has a higher flexibility than the optical scanner. It can accept answer sheets in different formats by simply changing the parameters related to the reference co-ordinates in the program, and if necessary, change the lens to suit the paper size.

Moreover, the scanner has a fixed computation time for forms with a few questions while this system has a shorter execution time for less questions. However, the image processing approach used in this design is sensitive to surrounding lighting. For example, any shadow cast on part of the answer sheet would lead to miss or false detection. This problem can be overcome by mounting a small lamp adjacent to the Imputer to provide a constant illumination over the sheet and eliminate shadow arising from other sources of light in the room.

This prototype can be used as a basis for later improved models such as an automatic system integrated with a paper feeding machine. Finally, whenever the marking system is not in use, the Imputer can be used for other applications.

REFERENCES

[1] Vellacott, O., "CMOS in Camera", IEE Review, Vol. 40, Iss. 3, pp.111-14, 1994.

[2] Sekita, I., Kurita, T., Otsu, N. and Abdelmalek, N., "A Thresholding Method using the Mixture of Normal Density Functions", ISSIPNN, Vol, 1, pp. 304-7, Hong Kong, April, 1994.

[3] Yin, P. Y. and Chen, L. H., "A New Method for Multilevel Thresholding using Symmetry and Duality of the Histogram", ISSIPNN, Vol. 1, pp. 45-8, Hong Kong, April, 1994.

[4] Shapiro, V. A., Veleva, P. K. and Sgurev, V. S., "An Adaptive Method for Image Thresholding", IEEE Proceeding on 11th IAPR International Conference on Pattern Recognition, Vol. 3, pp 696-699, Los Alamos, 1992.

[5] Parker, J. R., "Gray Level Thresholding in Badly Illuminated Images", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 13, Iss. 8, pp 813-819, USA, Aug 1991.

[6] Pratt, W. K., *"Digital Image Processing"*, Wiley, New York, 1991.

[7] Gonzalez, R. C. and Woods, R. E., *"Digital Image Processing"*, Addison-Wesley, New York, 1993.

[8] VLSI Vision, *"Imputer Release 2.5 User Manual"*, VLSI Vision, Edinburgh, 1994.

[9] VLSI Vision, *"Imputer Release 2.0 IP Library Reference"*, VLSI Vision, Edinburgh, 1994.

[10] VLSI Vision, *"C-Compiler User's Guide"*, VLSI Vision, Edinburgh, 1989.

[11] VLSI Vision, *"8051 Utilities User's Guide"*, VLSI Vision, Edinburgh, 1990.